IT-SecX 2011

Ähnlichkeitssuche bei Malware

DI Johannes Kaufmann

Agenda

- Die Problemstellung
- Kontext
- Das Modell
- Die Ähnlichkeitssuche
- Zusammenfassung und Ausblick

Problem: Erkennung von Malware

Malwaretechnik	Statische Analyse	Dynamische Analyse
Obfuskation des Codes	✘	✓
Verschlüsselung	✘	✓
Kompression	✘	✓
Antidebuggingmechanismen	✘	✓
Eigenständige Codeänderungen	✘	✓

Problemstellung

> Verstärkte Forschung im Bereich dynamischer Analyse

> Schwerpunkte:
  - Entwicklung von Sandboxen
  - Data Mining

> Wissen bleibt unberücksichtigt

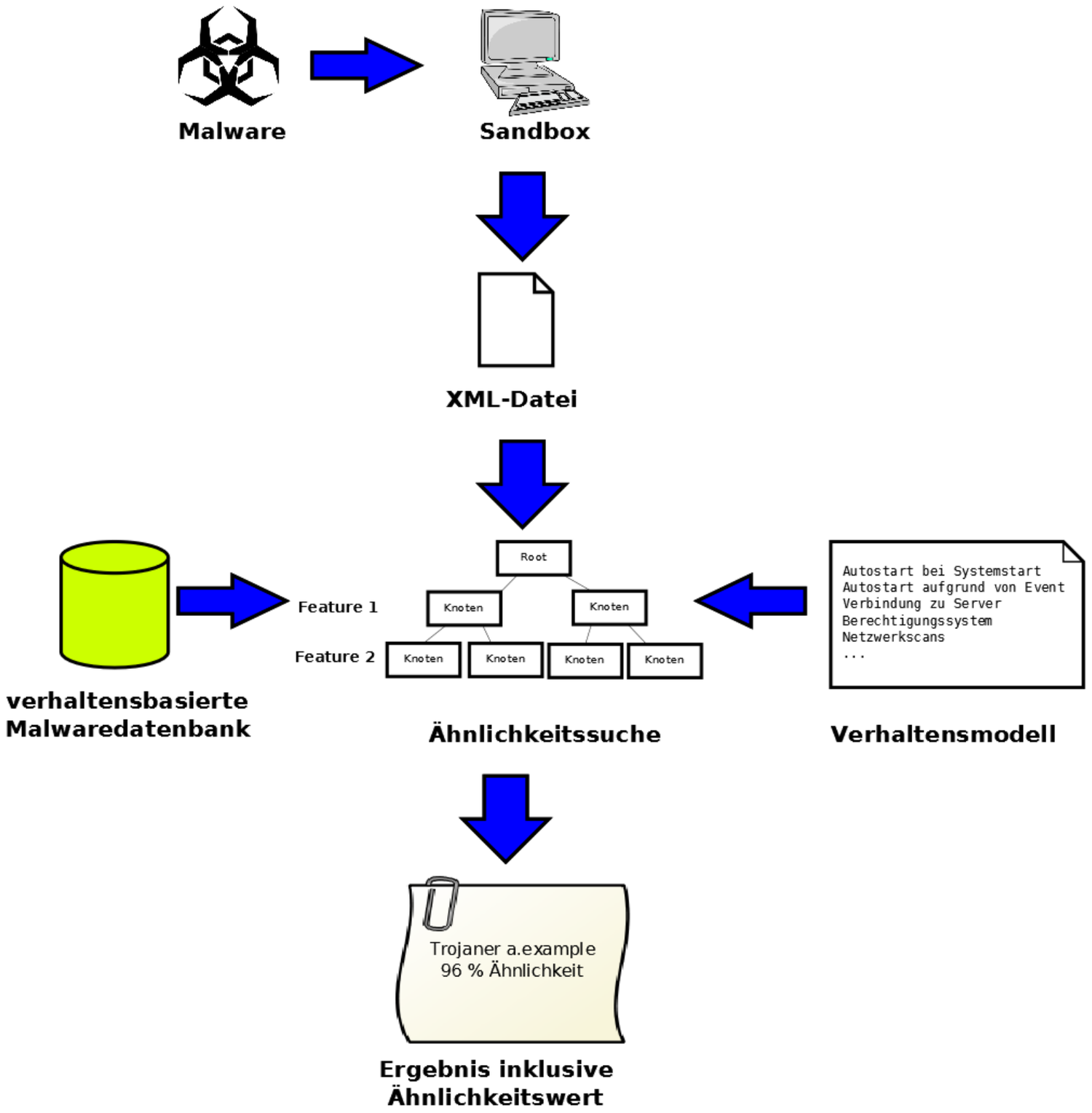
Problemstellung

- Ziel:
- Entwicklung eines Modells zur Darstellung von Malware
  - Möglichst generell
  - Kompakte Zahl an Features

- Ähnlichkeitssuche
  - Das Finden des dem Abfrageobjekts ähnlichsten Objekt in der Datenbank

Kontext

- Projektgruppe an der FH St. Pölten
- Infrastruktur:
  - ANUBIS-Sandbox
    - Windows XP
    - Derzeit kein Netzwerkzugang
  - Vorgefilterte XML-Dateien
  - Malware-Datenbank


Malware
Sandbox
XML-Datei
verhaltensbasierte
Malwaredatenbank
Feature 1
Feature 2
Root
Knoten
Knoten
Knoten
Knoten
Knoten
Ähnlichkeitssuche
Autostart bei Systemstart
Autostart aufgrund von Event
Verbindung zu Server
Berechtigungssystem
Netzwerkscans
...
Verhaltensmodell
Trojaner a.example
96 % Ähnlichkeit
Ergebnis inklusive
Ähnlichkeitswert

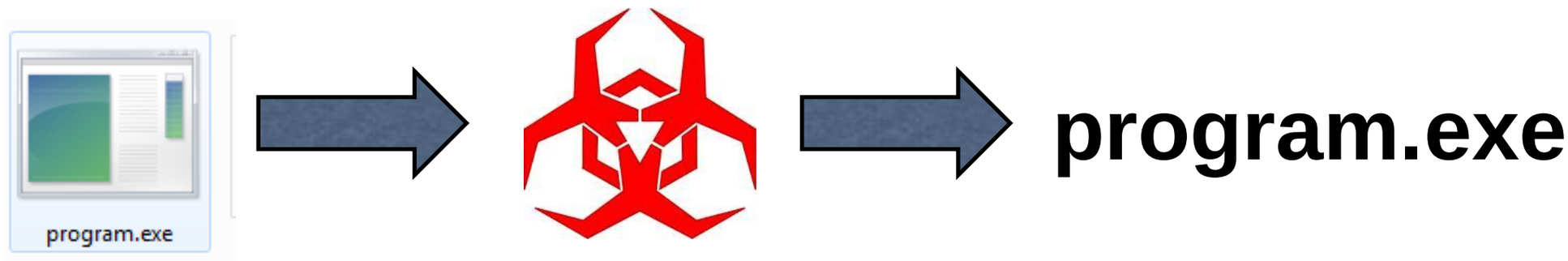
Das Malwaremodell

- 13 Features
- Gewichtung nach Signifikanz
- Spezifische Systemänderungen zugeordnet
- Formel zur Distanzberechnung abhängig von Feature
- Maximaldistanz vor Gewichtung: 100

Das Modell

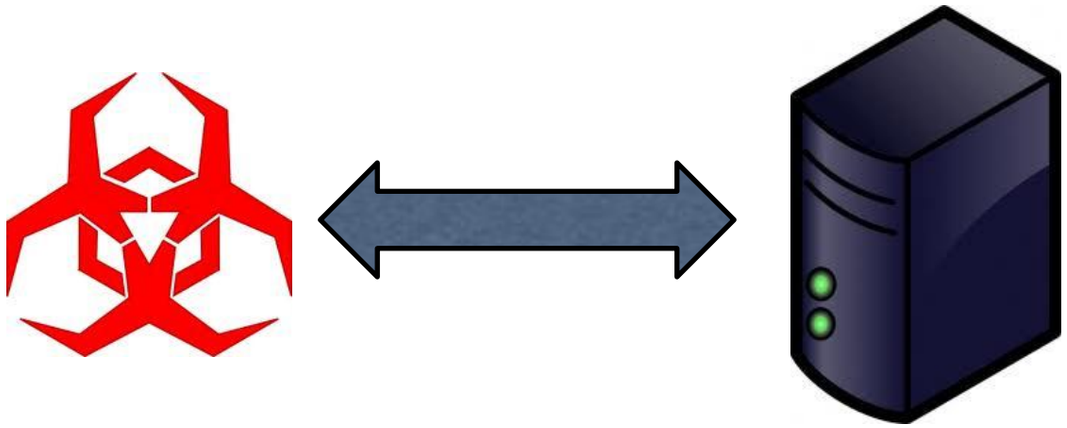
> 1. Autostart bei Systemstart

> 2. Autostart bei Ausführung von Dateitypen



Das Modell

> 3. Verbindungsversuch zu Kontrollserver



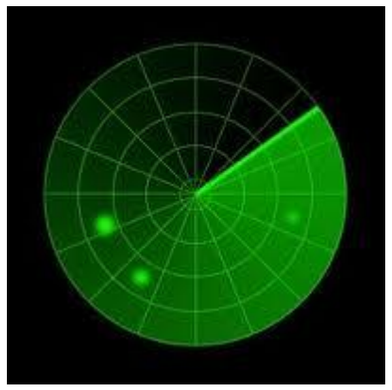
> 4. Modifizierung des Berechtigungssystems

Berechtigungseinträge:

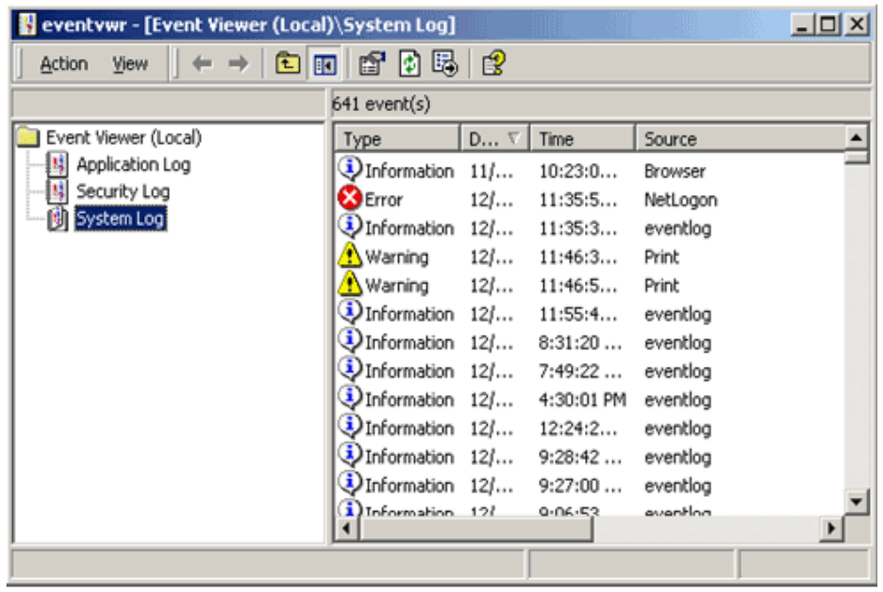
Typ	Name	Berechtigung
Zulassen	SYSTEM	Vollzugriff

Das Modell

> 5. Netzwerkscans



> 6. Änderung der Logeinträge



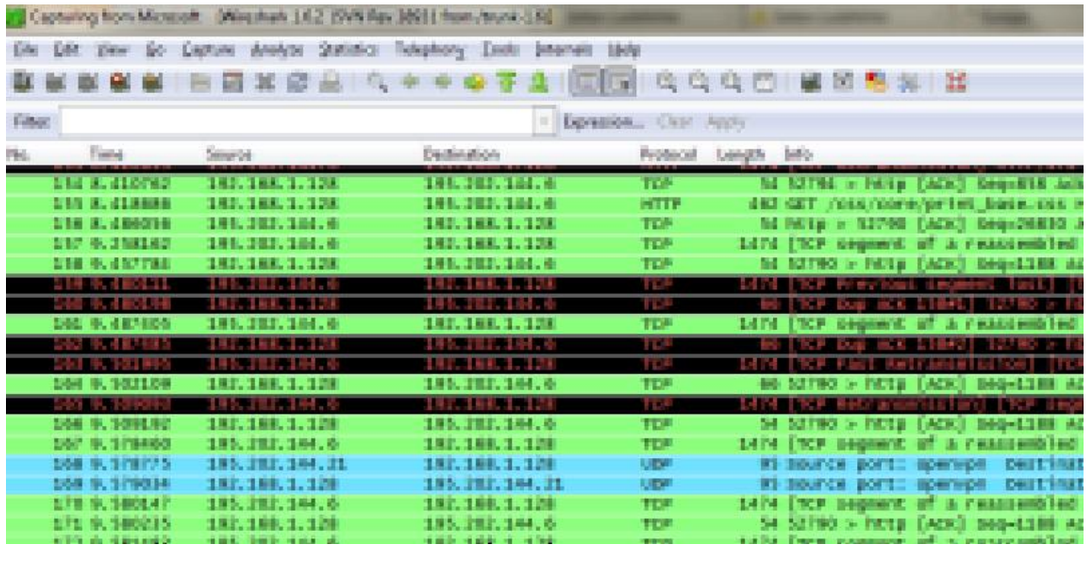
Das Modell

> 7. Firewall, Antivirenprogramm, Windows Security Center

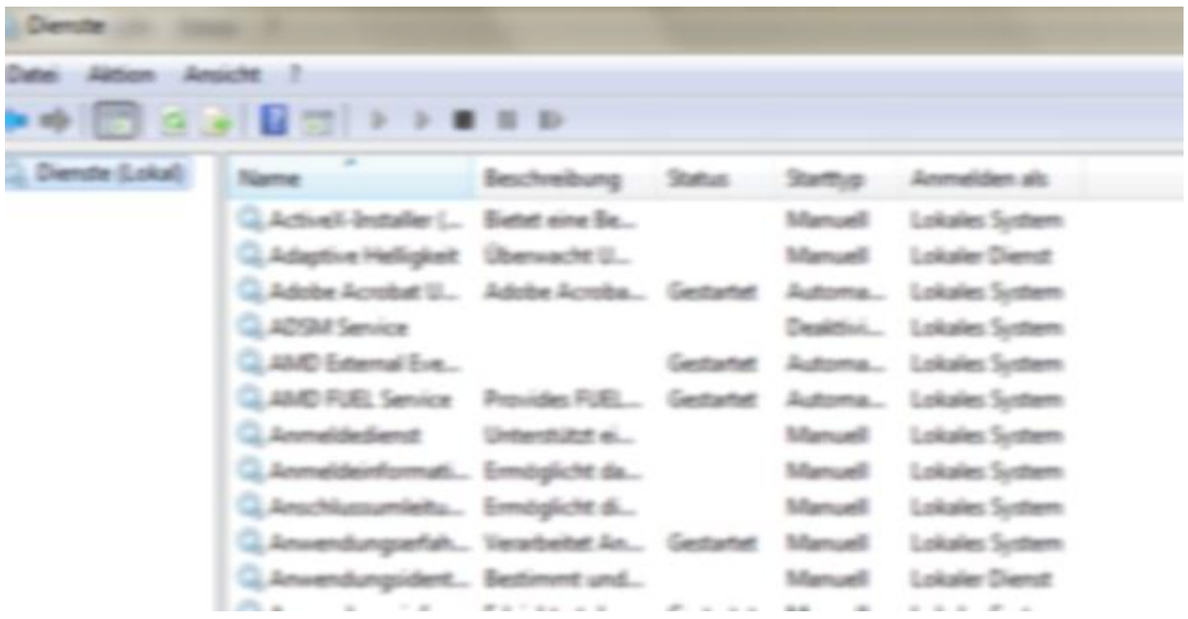
> 8. Weitere sicherheitsrelevante Einstellungen

Das Modell

> 9. Internet & Netzwerk



> 10. Services



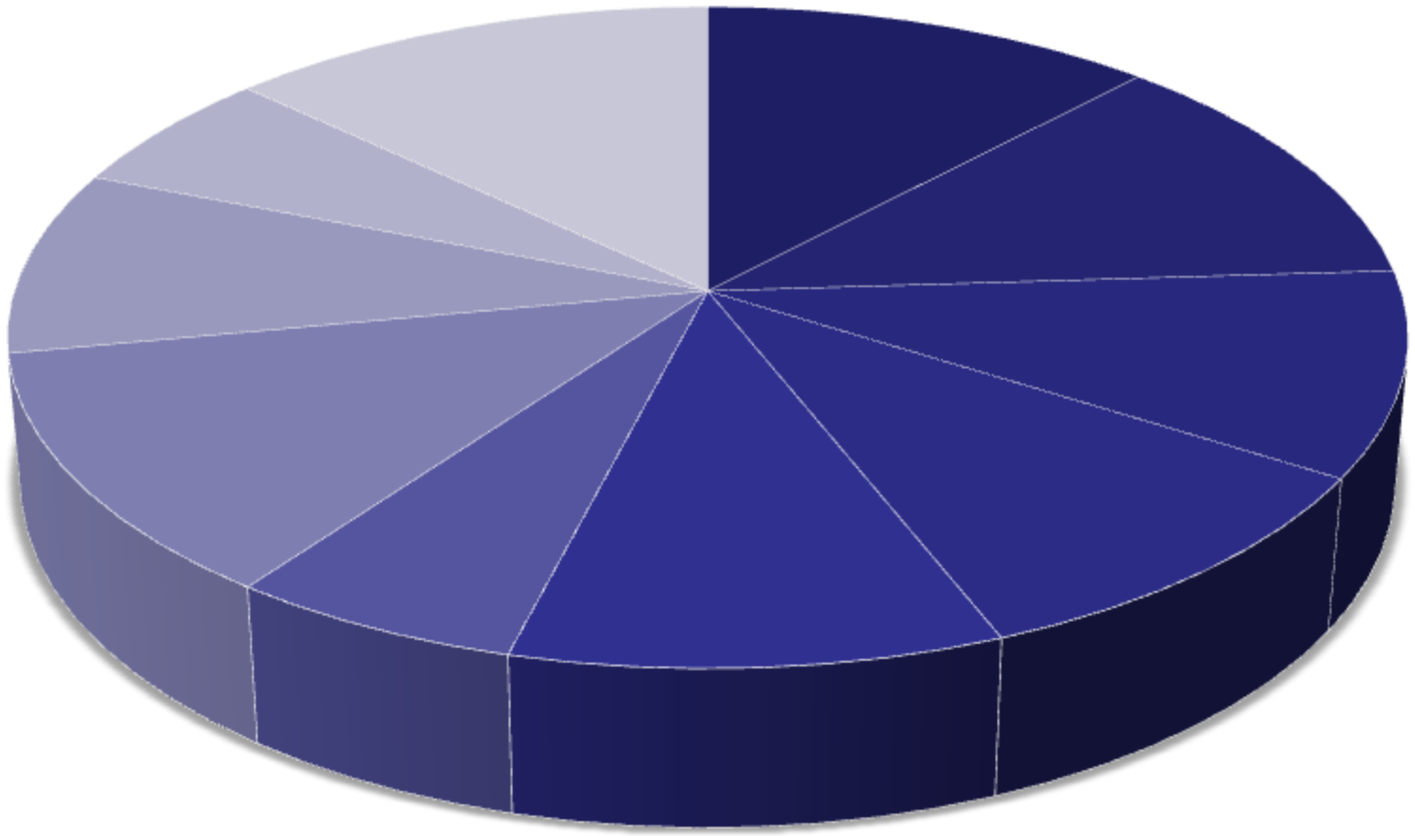
Das Modell

> 11. Änderung von Dateien

> 12. Prozesse

> 13. Registry Scan

Gewichtung der Kriterien



Die Ähnlichkeitssuche

> Anforderungen:

- Die dem Abfrageobjekt ähnlichsten Objekte aus einer Datenbank zu finden.
- Pro Attribut eine eigene Distanzfunktion
- Gewichtungen möglich

Der Attribute-Level-Tree

- Nicht heuristisch
- Besonders gut geeignet für „dichte“ Datenbanken
- Ausgabe der Ähnlichkeit in Prozent möglich
- Baumförmige Entscheidungsstruktur
- Pro Ebene ein Attribut

Der Attribute-Level Tree

> Aufbau eines Knotens:

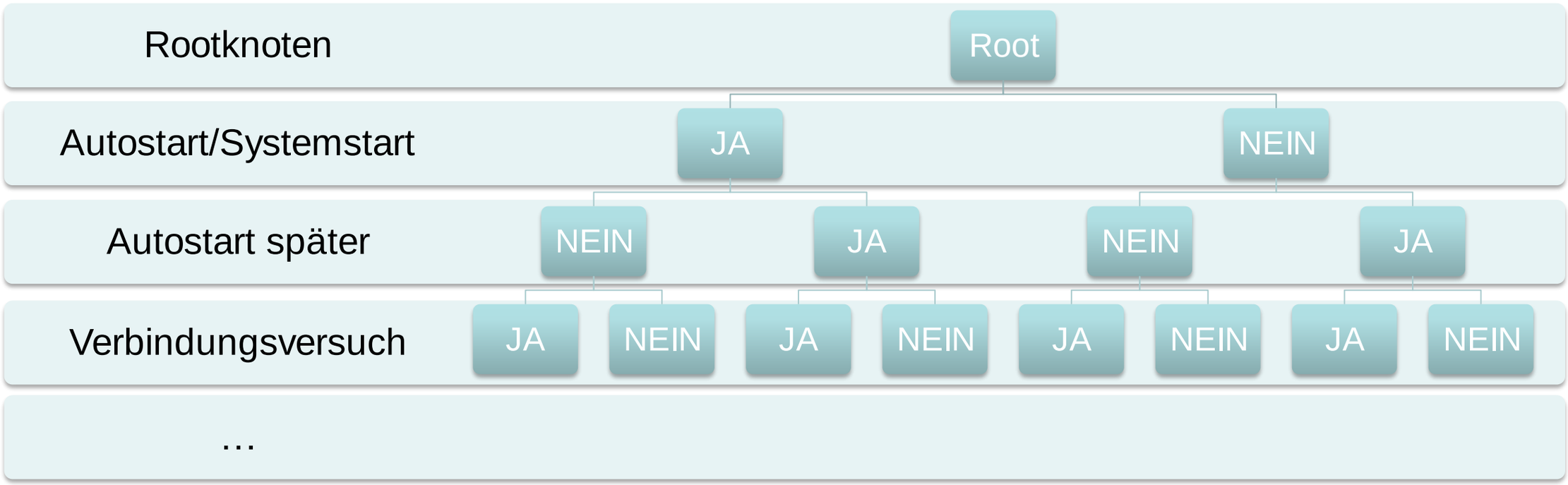
Arbeitsweise des ALT

1
- Kandidat auswählen
  - niedrigste Mindestdistanz

2
- Sortierung der Kinder
  - nach Ähnlichkeit

3
- Erstellung neuer Kandidaten
- Oder Eintragung in die Ergebnisliste

Der Attribute-Level Tree

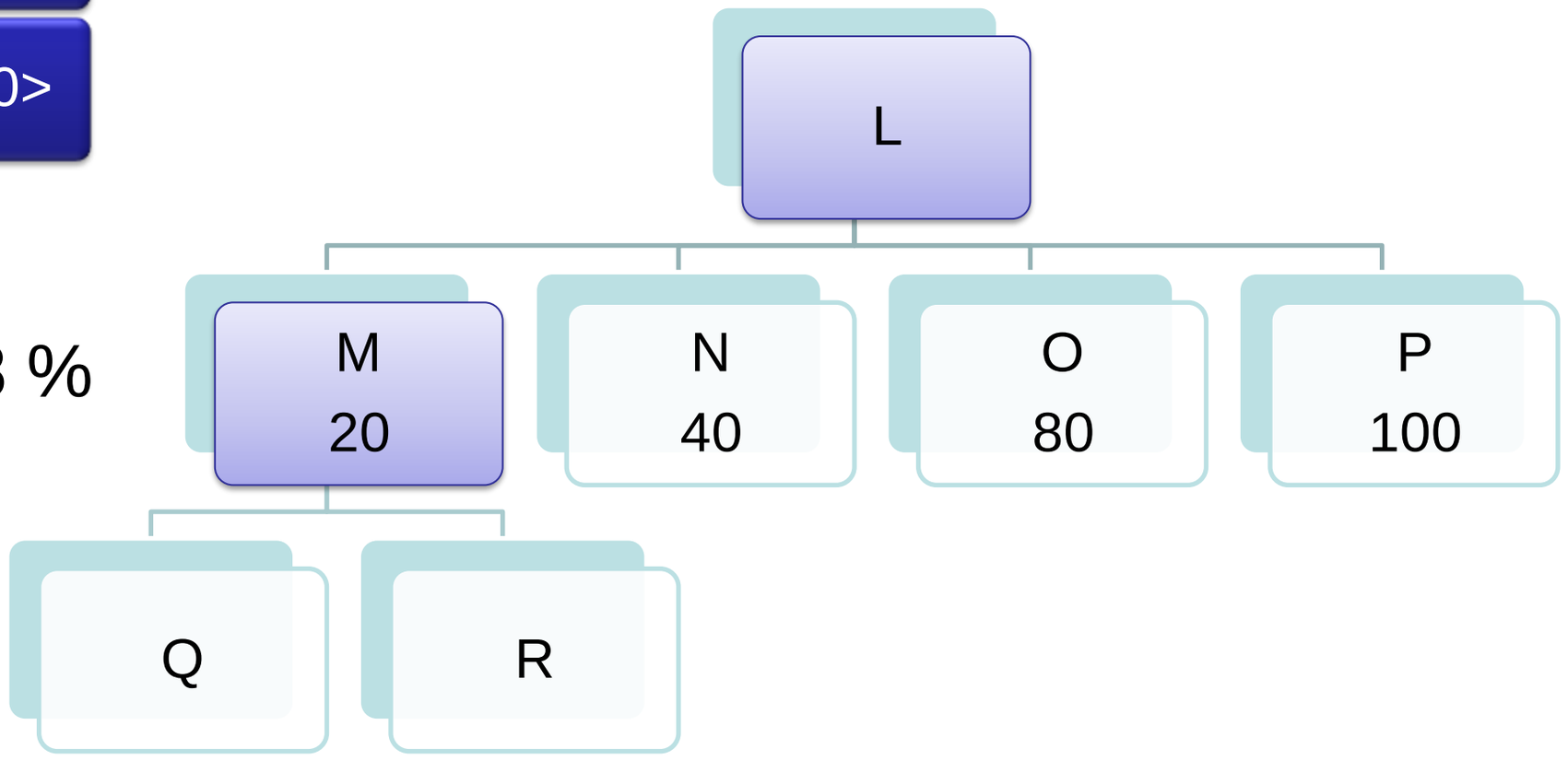


Der Attribute-Level Tree

Kandidatenliste

< L, M, 2300, 2400>

< S, T, 2700, 2700>

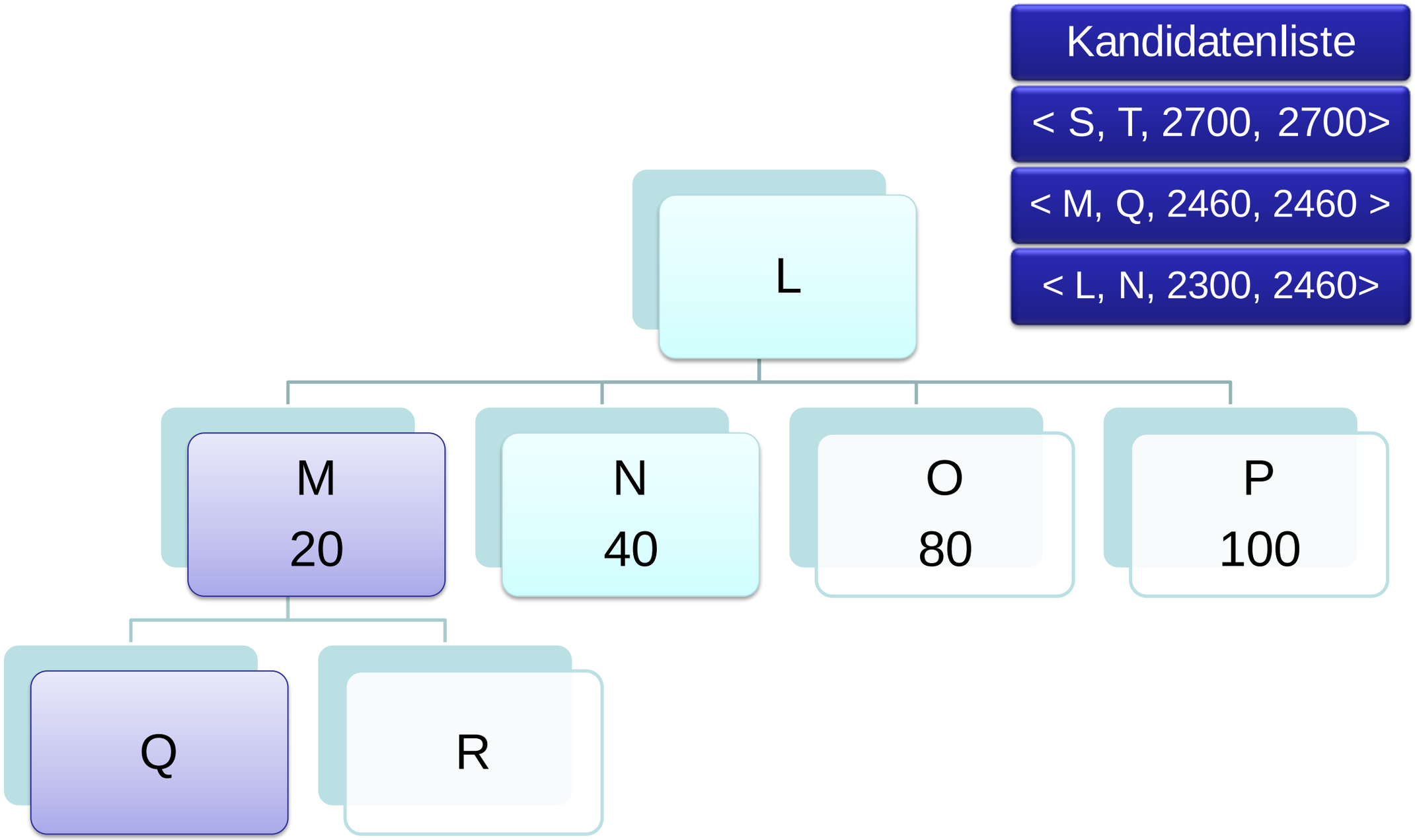


Gewichtung: 8 %

Der Attribute-Level Tree

- Expandierung < L, M, 2300, 2400>
- Berechnung der Distanz des Child-Knotens:
- 20 * 8 = 160
- Mindestdistanz = Distanz
- < M, Q, 2460, 2460>
- Erstellung der Variante:
- Distanz Variante = Distanz expandierter Knoten
- Mindestdistanz Variante = Mindestdistanz Child
- < L, N, 2300, 2460 >

Der Attribute-Level Tree



Conclusio und Ausblick

- Verhaltensmodell entwickelt
- Gut geeigneter Suchalgorithmus gefunden
- Implementation ausständig
  - Experimente
- Adaptierung des Modells leicht möglich
- Anpassung an modernere Betriebssysteme

IT-SecX 2011

Fragen und Antworten